A

jc520 U.S. PTO 07/14/99

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. **0100.9900940** Total Pages **31**
First Inventor or Application Identifier **Paul W. Campbell**
Title **Method and Apparatus for Virtual Address Translation**
Express Mail Label No. **EL280235073US**

jc542 U.S. PTO 09/352959 07/14/99

APPLICATION ELEMENTS *See MPEP chapter 600 concerning utility patent application contents.*	*ADDRESS TO:* **Assistant Commissioner for Patents Box Patent Application Washington, DC 20231**

1. ☒ Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. ☒ Specification *Total Pages* 18
(preferred arrangement set forth below)
- Descriptive title of the Invention
- Cross References to Related Applications
- Statement Regarding Fed sponsored R & D
- Reference to Microfiche Appendix
- Background of the Invention
- Brief Summary of the Invention
- Brief Description of the Drawings *(if filed)*
- Detailed Description
- Claim(s)
- Abstract of the Disclosure

3. ☒ Drawings *(35 USC 113) Total Sheets* 5
4. Oath or Declaration *Total Pages* 2
   a. ☒ Newly executed (original or copy)
   b. ☐ Copy from a prior application (37 CFR 1.63(d))
   (for continuation/divisional with Box 17 completed)
   [Note Box 5 below]
   i. ☐ DELETION OF INVENTOR(S)
   Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1 63(d)(2) and 1.33(b).

5. ☐ Microfiche Computer Program *(Appendix)*

6. ☐ Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. ☐ Computer Readable Copy
   b. ☐ Paper Copy (identical to computer copy)
   c. ☐ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

7. ☒ Assignment Papers (cover sheet & document(s))
8. ☐ 37 CFR 3.73(b) Statement (when there is an assignee) ☐ Power of Attorney
9. ☐ English Translation Document (if applicable)
10. ☐ Information Disclosure Statement (IDS)/PTO-1449 ☐ Copies of IDS Citations
11. ☐ Preliminary Amendment
12. ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
13. ☐ Small Entity Statement(s) ☐ Statement filed in Prior Application, Status still proper and desired.
14. ☐ Certified Copy of Priority Document(s)
(if foreign priority is claimed)
15. ☐ Other

16. If a CONTINUING APPLICATION, *check appropriate box and supply the requisite information:*
☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No:
Prior Application Information: Examiner *Group / Art Unit:*

17. CORRESPONDENCE ADDRESS

☐ *Customer Number or Bar Code Label* or, ☒ *Correspondence Address Below*

Markison & Reckamp, P.C.
175 West Jackson Boulevard - Suite 1015
Chicago, Illinois 60604
Telephone:312-939-9800 Facsimile: 312-939-9828

Name (Print/Type)	**Timothy W. Markison**	REGISTRATION NUMBER	**33,534**
Signature		Date	**7/14/99**

PATENT APPLICATION
DOCKET NO. 0100.9900940

In the United States Patent and Trademark Office

FILING OF A UNITED STATES PATENT APPLICATION

Title:

METHOD AND APPARATUS FOR VIRTUAL ADDRESS TRANSLATION

Inventors:

Paul W. Campbell **6652 Dana St.** **Oakland, CA 94609**	

Attorney of Record
Timothy W. Markison
Registration No. 33,534
175 W. Jackson Blvd. – Suite 1015
Chicago, Illinois 60604
Phone (512) 343-1103
Fax (512) 372-3990

EL280235073US

Express Mail Label No _____

Date of Deposit 7/14/99

I hereby certify that this paper is being deposited with the U S Postal Service "Express Mail Post Office to Addresses" service under 37 C F R Section 1 10 on the 'Date of Deposit', indicated above, and is addressed to the Commissioner of Patents and Trademarks, Washington, D C 20231

Name of Depositor TIM MARKISON
(print or type)

Signature Tim Mark

Express Mail Label No. EL280235073US

METHOD AND APPARATUS FOR VIRTUAL ADDRESS TRANSLATION

Technical Field of the Invention

This invention relates generally to computer architectures and more particularly to virtual, or linear, address translations.

Background of the Invention

Figure 1 illustrates a schematic block diagram of a portion of a computer system. As shown, a central processing unit is coupled to cache memory and to a north bridge. The north bridge is coupled to memory, an accelerated graphics port (AGP) bus, and a PCI bus. The central processing unit addresses memory in system virtual address space, or linear address space. To communicate with the north bridge, the central processing unit converts addresses in virtual address space to addresses in physical address space. To make such a conversion, the central processing unit often utilizes page address translation and includes a translation look-aside table (TLB) for storing the conversions.

The north bridge, upon receiving an address in physical address space from the central processing unit, determines whether the address corresponds to memory, PCI address space, or AGP memory space. If the address is directed towards the AGP address space, the north bridge makes a further translation of the received physical address utilizing a GART translation. The translated address is then stored in a GART TLB. As such, for the central processing unit to communicate with the AGP bus, two

address space translations occur utilizing separate paging, page address translations and two separate TLBs are maintained.

Figure 2 illustrates a graphical representation of address space within the system of Figure 1. As shown, the system virtual address space, which corresponds to the central processing unit, has memory space for input/output transactions, kernels and processes. The system virtual addresses are converted to physical addresses as previously discussed. The physical address space includes PCI address space, AGP address space, main memory address space, and DOS address space. The main memory address space corresponds to the DRAM address space, while the PCI address space corresponds to the PCI address space along the PCI bus. As shown, for the AGP address space needs to be converted through a GART translation and is stored in main memory. Hence the need for the extra conversion.

Figure 3 illustrates a logic diagram of a method for address translations. The process begins with the central processing unit determining whether an address has a corresponding translation in its TLB. If not, the CPU indexes, based on a portion of the linear address, a page directory to obtain a page directory entry (PDE). The central processing unit then indexes a page table based on the PDE and another portion of the linear address to obtain a page table entry (PTE). At this point, the central processing unit obtains the physical address based on the page table entry and yet another portion of the virtual, or linear, address. In a typical 32 bit virtual address, the first ten bits are used to address the page directory, the next ten bits are used to address the page table and the remaining bits correspond the least significant bits of the physical address. Note that if the translation has been stored in the TLB, the central processing unit may retrieve the physical address directly from the TLB.

Having obtained the physical address, the central processing unit determines whether the data is cached. If so, the process is done for this particular address. If not, the central processing unit passes the physical address to the north bridge. Upon receiving the physical address, the north bridge determines whether the physical address

is in the AGP window. If not, the north bridge causes the corresponding data to be read from memory and sent to the central processing unit. If, however, the physical address is in the AGP window, the north bridge determines whether an AGP translated address is stored in a GART TLB. If so, the translated address is used to retrieve data from memory, which data is subsequently provided to the central processing unit. If, however, a GART TLB entry is not stored, the north bridge performs an AGP translation using a GART table.

As such when addresses produced by the central processing unit correspond to the AGP window, two translations occur. In existing computer systems, the two translations are done by the central processing unit and the north bridge. As such, each device contains a separate TLB. Such redundancy adds extra processing steps to address translations and produces overlapping data storage. In addition, it is more difficult to cache data from memory within the AGP window because the second address translation occurs within the north bridge.

Therefore, a need exists for a method and apparatus that more efficiently performs virtual address translations.

Brief Description of the Drawing

Figure 1 illustrates a schematic block diagram of a portion of a prior art computing system;

Figure 2 illustrates a graphic representation of address space in a prior art computing system;

Figure 3 illustrates a logic diagram of a method for address space conversion in a prior art computing system;

Figure 4 illustrates a schematic block diagram of a computing system in accordance with the present invention;

Figure 5 illustrates a logic diagram of a method for address space translation in accordance with the present invention; and

Figure 6 illustrates a logic diagram of a method for generating a translation look aside table entry in accordance with the present invention.

Detailed Description of a Preferred Embodiment

Generally, the present invention provides a method and apparatus for virtual address translation. Such a method and apparatus include processing that begins by receiving a memory access request that includes a virtual address. The processing continues by determining whether a physical address translation has been performed for the virtual address. Note that a physical address translation translates the virtual address (often referred to as a linear address) into an address. The address either corresponds to physical address of memory or is further translated into another physical address of memory. The processing continues when the address, which resulted from the physical address translation or the another physical address translation, is stored in a translation look aside table (TLB). When the physical address translation or the another physical address translation has not been performed, the processing retrieves a physical page address based on a portion of the virtual address. The processing continues by determining whether the physical page address corresponds to a physical address requiring further translation (e.g., is within the AGP window). If not, the physical page address and a portion of the virtual address are used to produce the physical address. If the address is within the physical address requiring further translation, a second physical page address is retrieved. Utilizing the second physical page address and a portion of the virtual address, another physical address is produced. Both types of translations (i.e., the physical address and the another physical address) are stored within the same TLB of the central processing unit. With such a method and apparatus, address translations into and

from the AGP window are performed with less operational instructions, are performed within the CPU, are performed without redundant or separate TLBs, and allows for ease in caching data from memory within the AGP window.

The present invention can be more fully described with reference to Figures 4-6. Figure 4 illustrates a schematic block diagram of a computing system 10 that includes a pair of central processing units 12 and 14, a memory gateway 16, an I/O (input/output) gateway 18, cache memory 20 and 22, and memory 24. Each of the central processing units includes a separate translation look aside table (TLB) 30 and 32. The central processing units 12 and 14, the memory gateway 16, the IO gateway 18 and the cache memories 20 and 22 are operably coupled to a global bus such that all memory access requests on the bus are done utilizing the same address space. Typically, the address space will be the physical address space but could be a virtual address space. Note that the I/O gateway 18 provides coupling to the PCI bus 26 and to the AGP bus 28. As such, the I/O gateway 18 performs the address space translation from PCI space to the address space used on the global bus. In addition, the I/O gateway 18 enables data to be read and/or written over the AGP bus, however, the AGP address translation is done within the CPU as described herein. The memory gateway 16 performs the address translation from the address space on the bus to physical address space.

To facilitate address space conversions, the central processing units 12 and 14 utilize their respective TLBs 30 and 32 to perform the methods shown in Figures 5 and 6. The processing of Figure 5 begins at step 40 where a memory access request is received, wherein the memory access request includes a virtual address (a.k.a., a linear address). The process then proceeds to step 42 where a determination is made as to whether a physical address translation has been performed for a virtual address. In one embodiment, the central processing unit processing this particular memory request by accessing its TLB and determining whether an entry exists for this particular virtual address. If so, the process proceeds to step 44 where the physical address or another physical address is used to obtain data which corresponds to the memory access requests. The physical address corresponds to a single translation of the virtual address while

another physical address corresponds to the virtual address undergoing two translations wherein the another physical address corresponds to the AGP window of main memory.

If a physical address translation has not been performed for the virtual address, the process proceeds to step 46. At step 46 a physical page address is retrieved. The physical page address is retrieved utilizing a first portion of the virtual address via known paging techniques involving page directories and page tables. The process then proceeds to step 48 where a determination is made as to whether the physical page address corresponds to a physical address requiring further translation. In one embodiment, the determination at step 48 is determining whether the physical address corresponds to the AGP window. As one of average skill in the art would appreciate, other portions of the main memory may be reserved for other applications, which would require a further translation similar to addresses corresponding to the AGP window.

If the physical page address does not correspond to a physical address requiring further translation, the process proceeds to step 50. At step 50, the physical page address and a portion of the virtual address are used to produce the physical address. The process then proceeds to step 52 where the physical address, the physical page address, and/or the portion of the virtual address are stored in the CPUs translation look aside table. The process then proceeds to step 44 where the TLB entry is accessed to retrieve the corresponding physical address such that the appropriate data is obtained.

If, however, the physical page address does correspond to a physical address requiring further translation, the process proceeds to step 54. At step 54, a second physical page address is retrieved utilizing the first portion of the physical address requiring further translation via known paging techniques involving page directories and page tables. This second physical page address may correspond to AGP window space and may be the equivalent of an entry in a GART table. The process then proceeds to step 56 where the second physical page address and a portion of the virtual address are used to produce another physical address. The process then proceeds to step 58 where another physical address, the second physical page address, and/or a portion of the virtual

address are stored in the CPUs TLB. The TLB is then accessed to retrieve the another physical address such that the corresponding data may be obtained at step 44.

The obtaining of data at step 44 may be further described with steps 60 and 62. At step 62, a determination is made as to whether the data is cached. If so, the data is retrieved and the process is done. If not, the process proceeds to step 62 where the physical address or the another physical address is provided to the memory gateway to retrieve the requested data.

Figure 6 illustrates a logic diagram for generating translation look aside table entries in accordance with the present invention. Such processing begins at step 70 where a virtual address is translated into an address. This may be performed in accordance with steps 80 and 82. At step 80, a page directory is indexed based on a first portion of the virtual address to retrieve a page directory entry. The process then proceeds to step 82 where a page table is indexed based on the page directory and a second portion of the virtual address to retrieve a page table entry, which forms part of the address.

Returning to the main flow, the process proceeds to step 72 where a determination is made as to whether the address corresponds to translation memory space. One such translation memory space may be the AGP window, but could be any addressing that is further translated from the physical memory address. If not, the process proceeds to step 74 where the address, the page table entry, and the portion of the virtual address are stored in the translation look aside table.

If the address does correspond to a translation memory space, the process proceeds to step 76. At step 76 the address is translated into another address. This may be further described in accordance with step 84 where at least the page table entry and a third portion of the virtual address are translated into the another address which is within the video graphics memory space. Returning to the main flow, the process proceeds to step 78 where the another address is cached in the TLB. As such the address and the

another address paging translations are stored within the same TLB of the central processing unit. The caching of addresses within the TLB may be further defined with reference to step 86. At step 86 the page directory entry, the page table entry, and the third portion of the virtual address are cached as the address in the TLB.

The preceding discussion has presented a method and apparatus for virtual address translations. With such a method and apparatus, the central processing unit can perform both physical address translations and physical address translations into the AGP window space without the need for the north bridge. As such, a single TLB may be utilized to perform such address translations thereby reducing the complexity, redundancy, and extra processing required in previous computer architectures. In addition, the single TLB processing allows for ease in caching data from memory within the AGP window.

Claims

What is claimed is:

1. A method for virtual address translation, the method comprises the steps of:

a) receiving a memory access request that includes a virtual address;

b) determining whether a physical address translation has been performed for the virtual address, wherein the physical address translation translates the virtual address to an address, wherein the address is a physical address of memory or is further translated to obtain another physical address of the memory; and

c) when the physical address translation or the another physical address translation has been performed for the virtual address, utilizing the physical address or the another physical address to obtain data corresponding to the memory access request.

2. The method of claim 1 further comprises:

when a physical address translation has not been performed for the virtual address, retrieving a physical page address;

determining whether the physical page address corresponds to a physical address requiring further translation;

when the physical page address does not correspond to a physical address that requires further translation, utilizing the physical page address and a portion of the virtual address to produce the physical address.

3. The method of claim 2 further comprises caching the physical address in a translation look aside table.

4. The method of claim 3 further comprises:

when the physical page address corresponds to a physical address that requires further translation, retrieving a second physical page address; and

utilizing the second physical page address and a portion of the virtual address to produce the another physical address.

5. The method of claim 4 further comprises caching the another physical address in the translation look aside table.

6. The method of claim 5 further comprises when data corresponding to the memory access request is cached, utilizing the data.

7. A method for virtual address translation, the method comprises the steps of:

a) translating a virtual address into an address;

b) determining whether the address corresponds to translation memory space;

c) caching the address in a translation look aside table when the address does not correspond to the translation memory space;

d) translating the address into another address when the address corresponds to translation memory space; and

e) caching the another address in the translation look aside table.

8. The method of claim 7, wherein step (a) further comprises:

indexing a page directory based on a first portion of the virtual address to retrieve a page directory entry; and

indexing a page table based on the page directory and a second portion of the virtual address to retrieve a page table entry as at least part of the address.

9. The method of claim 8, wherein step (b) further comprises determining whether the page table entry is in video graphics memory space.

10. The method of claim 9, wherein step (c) further comprises caching the page directory entry, the page table entry, and a third portion of the virtual address as the address.

11. The method of claim 8, wherein step (d) further comprises translating at least the page table entry and a third portion of the virtual address into an address within the video graphics memory space.

12. A virtual address translation module comprises:

a processing module; and

memory operably coupled to the processing module, wherein the memory stores operational instructions that cause the processing module to: (a) receive a memory access request that includes a virtual address; (b) determine whether a physical address translation has been performed for the virtual address, wherein the physical address translation translates the virtual address to an address, wherein the address is a physical address of memory or is further translated to obtain another physical address of the memory; and (c) utilize the physical address or the another physical address to obtain data corresponding to the memory access request when the physical address translation or the another physical address translation has been performed for the virtual address.

13. The virtual address translation module of claim 12, wherein the memory further comprises operational instructions that cause the processing module to:

retrieve a physical page address when a physical address translation has not been performed for the virtual address;

determine whether the physical page address corresponds to a physical address requiring further translation;

utilize the physical page address and a portion of the virtual address to produce the physical address when the physical page address does not correspond to a physical address that requires further translation.

14. The virtual address translation module of claim 13, wherein the memory further comprises operational instructions that cause the processing module to cache the physical address in a translation look aside table.

15. The virtual address translation module of claim 14, wherein the memory further comprises operational instructions that cause the processing module to:

retrieve a second physical page address when the physical page address corresponds to a physical address that requires further translation; and

utilize the second physical page address and a portion of the virtual address to produce the another physical address.

16. The virtual address translation module of claim 15, wherein the memory further comprises operational instructions that cause the processing module to: cache the another physical address in the translation look aside table.

17. The virtual address translation module of claim 16, wherein the memory further comprises operational instructions that cause the processing module to utilizing the data when data corresponding to the memory access request is cached.

18. A virtual address translation module comprises:

a processing module; and

memory operably coupled to the processing module, wherein the memory stores operational instructions that cause the processing module to: (a) translate a virtual address into an address; (b) determine whether the address corresponds to translation memory space; (c) cache the address in a translation look aside table when the address does not correspond to the translation memory space; (d) translate the address into another address when the address corresponds to translation memory space; and (e) cache the another address in the translation look aside table.

19. The virtual address translation module of claim 18, wherein the memory further comprises operational instructions that cause the processing module to translate the virtual address by:

indexing a page directory based on a first portion of the virtual address to retrieve a page directory entry; and

indexing a page table based on the page directory and a second portion of the virtual address to retrieve a page table entry as at least part of the address.

20. The virtual address translation module of claim 19, wherein the memory further comprises operational instructions that cause the processing module to determine whether the page table entry is in video graphics memory space.

21. The virtual address translation module of claim 20, wherein the memory further comprises operational instructions that cause the processing module to cache the page directory entry, the page table entry, and a third portion of the virtual address as the address.

22. The virtual address translation module of claim 21, wherein the memory further comprises operational instructions that cause the processing module to translate at least the page table entry and a third portion of the virtual address into an address within the video graphics memory space.

METHOD AND APPARATUS FOR VIRTUAL ADDRESS TRANSLATION

Abstract of the Disclosure

A method and apparatus for virtual address translation include processing that begins by receiving a memory access request that includes a virtual address. The processing continues by determining whether a physical address translation has been performed for the virtual address. Note that a physical address translation translates the virtual address into an address. The address either corresponds to physical address of memory or is further translated into another physical address of memory. The processing continues when the address, which resulted from the physical address translation or the another physical address translation, is stored in a translation look aside table (TLB). When the physical address translation or the another physical address translation has not been performed, the processing retrieves a physical page address based on a portion of the virtual address. The processing continues by determining whether the physical page address corresponds to a physical address requiring further translation (e.g., is within the AGP window). If not, the physical page address and a portion of the virtual address are used to produce the physical address. If the address is within the physical address requiring further translation, a second physical page address is retrieved. Utilizing the second physical page address and a portion of the virtual address, another physical address is produced. Both the translations (i.e., the physical address and the another physical address) are stored within the same TLB of the central processing unit.

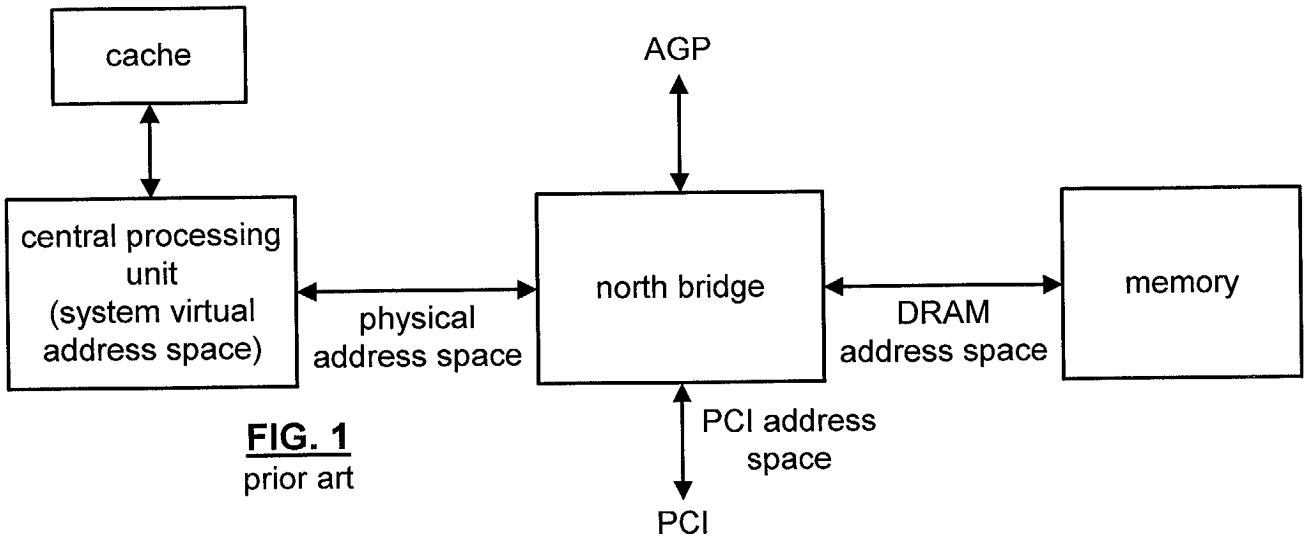

FIG. 1
prior art

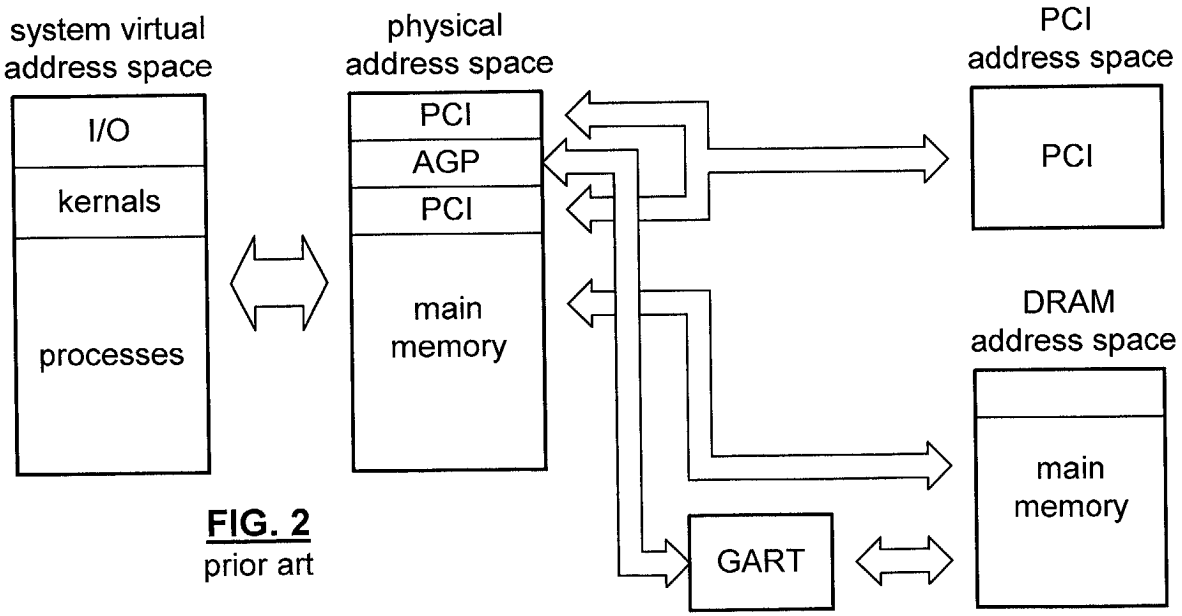

FIG. 2
prior art

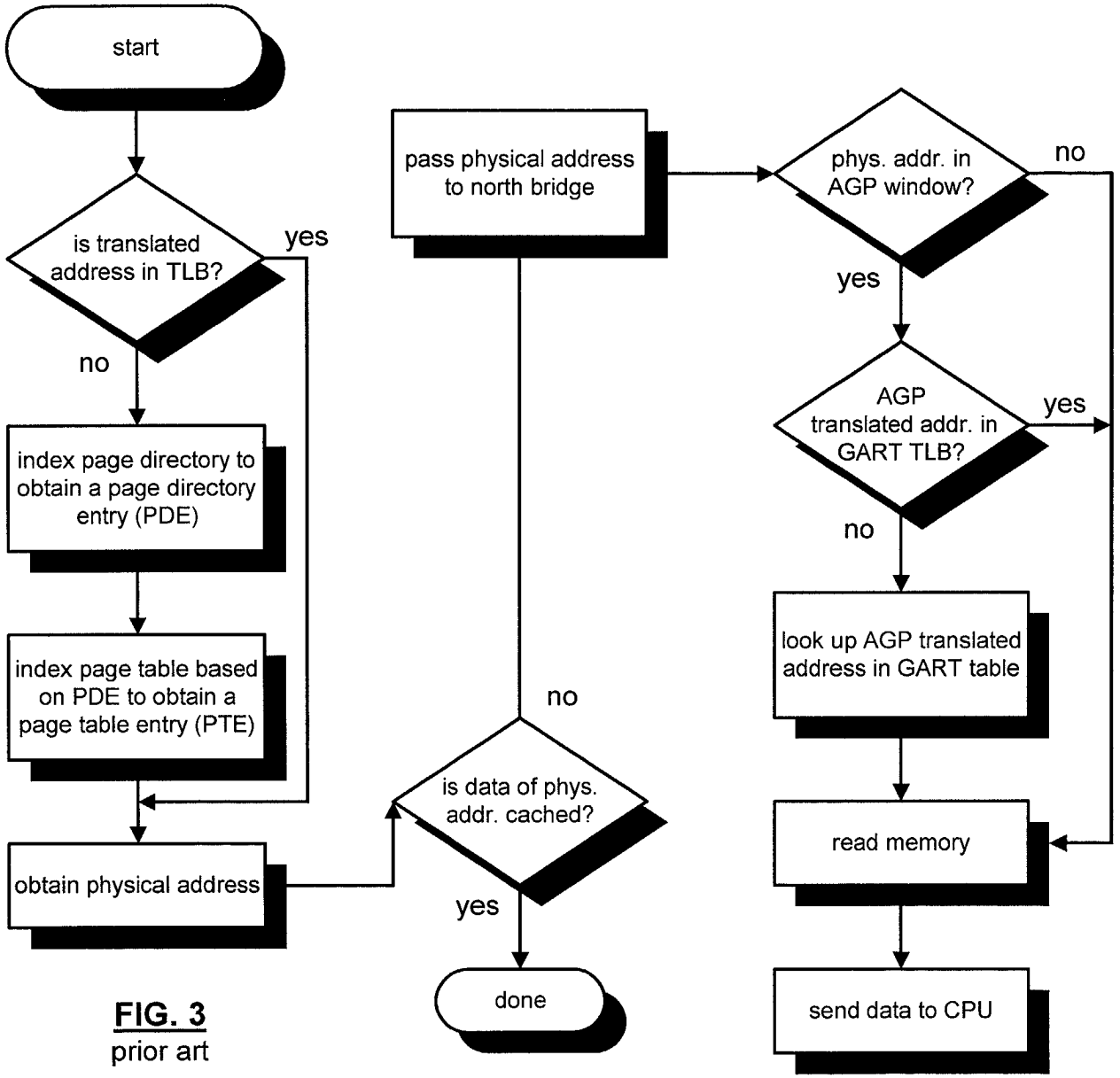

FIG. 3
prior art

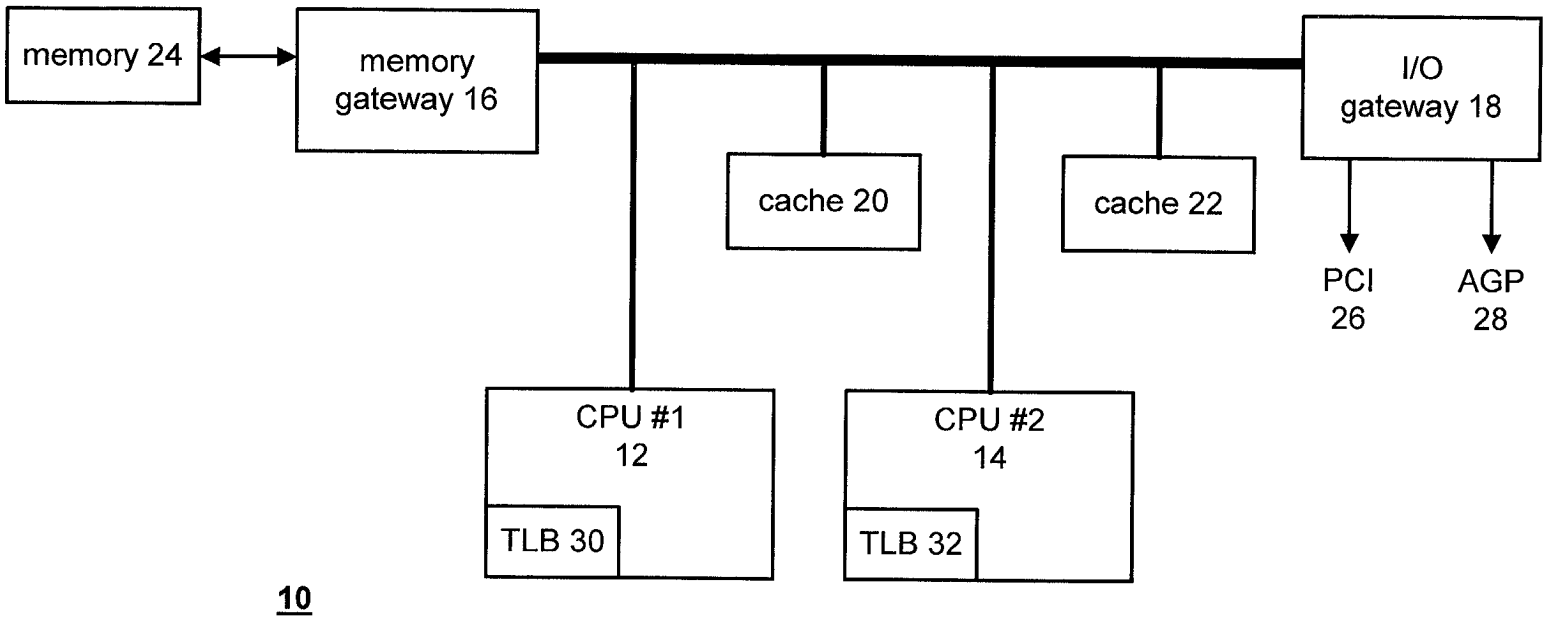
memory 24
memory gateway 16
I/O gateway 18
cache 20
cache 22
PCI 26
AGP 28
CPU #1 12
TLB 30
CPU #2 14
TLB 32
10

FIG. 4

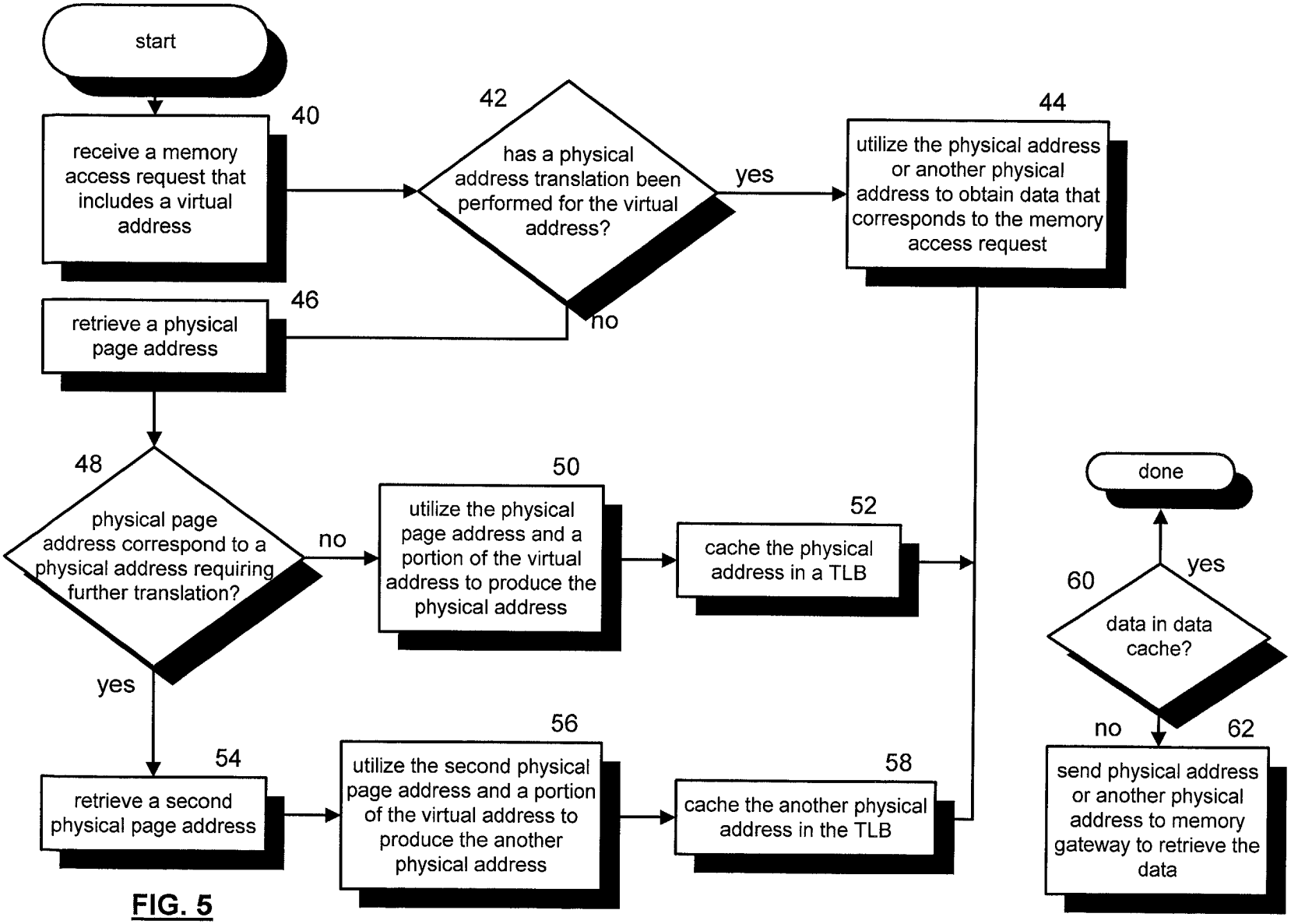
start
40
receive a memory access request that includes a virtual address
42
has a physical address translation been performed for the virtual address?
yes
44
utilize the physical address or another physical address to obtain data that corresponds to the memory access request
no
46
retrieve a physical page address
48
physical page address correspond to a physical address requiring further translation?
no
50
utilize the physical page address and a portion of the virtual address to produce the physical address
52
cache the physical address in a TLB
yes
54
retrieve a second physical page address
56
utilize the second physical page address and a portion of the virtual address to produce the another physical address
58
cache the another physical address in the TLB
done
60
data in data cache?
yes
no
62
send physical address or another physical address to memory gateway to retrieve the data

FIG. 5

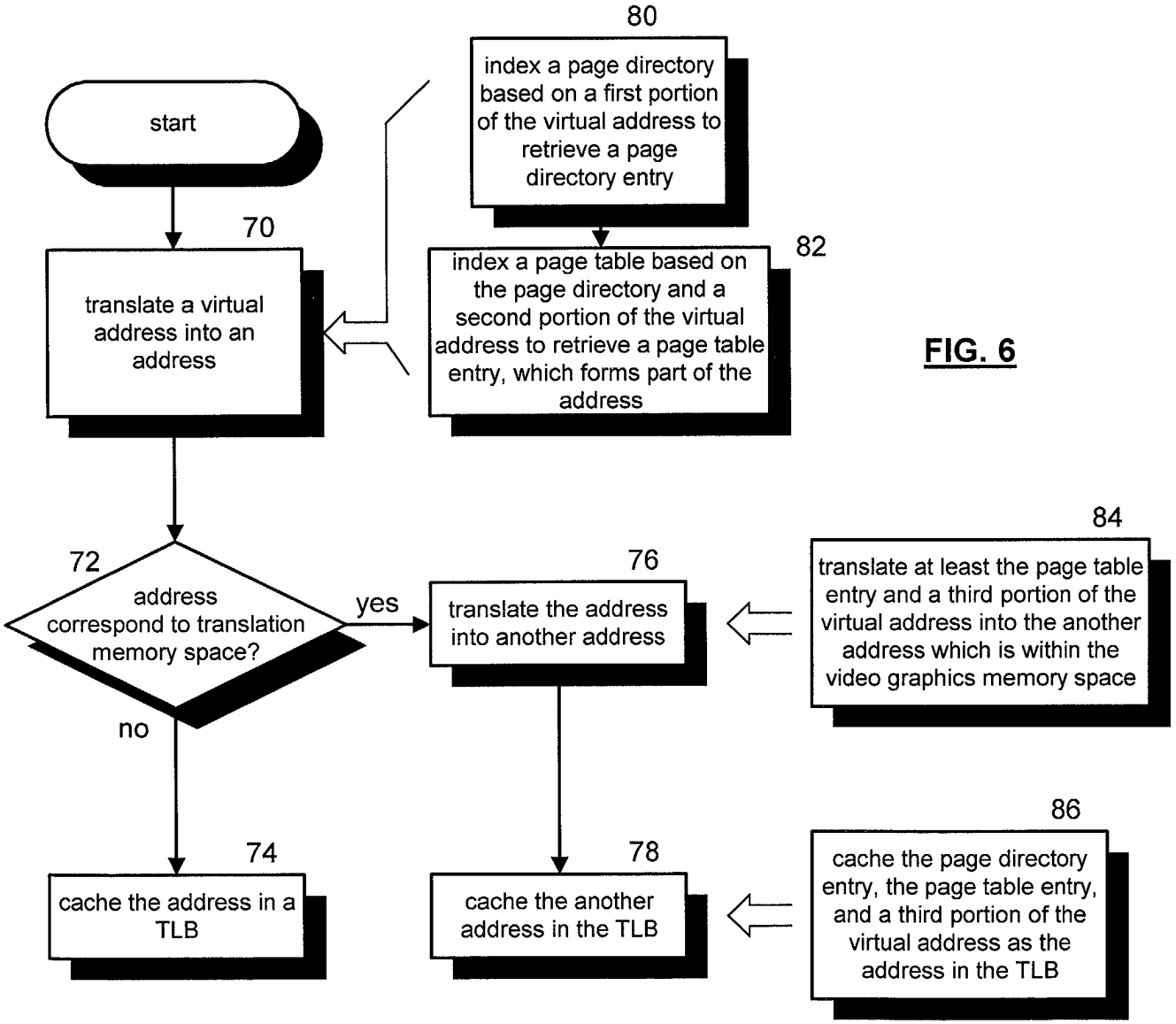
start
70
translate a virtual address into an address
80
index a page directory based on a first portion of the virtual address to retrieve a page directory entry
82
index a page table based on the page directory and a second portion of the virtual address to retrieve a page table entry, which forms part of the address
72
address correspond to translation memory space?
yes
no
76
translate the address into another address
84
translate at least the page table entry and a third portion of the virtual address into the another address which is within the video graphics memory space
74
cache the address in a TLB
78
cache the another address in the TLB
86
cache the page directory entry, the page table entry, and a third portion of the virtual address as the address in the TLB

FIG. 6

DECLARATION FOR UTILITY OR DESIGN PATENT APPLICATION (37 CFR 1.63)

☒ Declaration Submitted with Initial Filing, OR
☐ **Declaration Submitted after Initial Filing (surcharge (37 CFR 1.16 (e)) required)**

Attorney Docket Number 0100.9900940
First Named Inventor Paul W. Campbell
COMPLETE IF KNOWN
Application Number
Filing Date
Group Art Unit
Examiner Name

As a below named inventor, I hereby declare that:
My residence, post office address, and citizenship are as stated below next to my name.
I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled: **METHOD AND APPARATUS FOR VIRTUAL ADDRESS TRANSLATION**
the specification of which:
☒ is attached hereto.
☐ was file on (MM/DD/YYYY) as United States Application Number or PCT International Application Number and was amended on (MM/DD/YYYY) (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment specifically referred to above.
I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56.

I hereby claim foreign priority benefits under 35 U S C 119(a)-(d) or 365(b) of any foreign application(s) for patent or inventor's certificate, or 365(a) of any PCT international application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or of any PCT international application having a filing date before that of the application on which priority is claimed

Prior Foreign Application Number(s)	Country	Foreign Filing Date (MM/DD/YYYY)	Priority Not Claimed	Certified Copy Attached? YES	NO
			☐	☐	☐
			☐	☐	☐

☐ Additional foreign application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto

I hereby claim the benefit under 35 U.S.C 119(e) of any United States provisional application(s) listed below.

Application Number(s)	Filing Data (MM/DD/YYYY)

☐ Additional provisional application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto

I hereby claim the benefit under 35 U.S C. 120 of any United States application(s), or 365(c) of any PCT international application designating the United States of America, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U S C 112, I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1 56 which became available between the filing date of the prior application and the national or PCT international filing date of this application

U.S. Parent Application or PCT Parent Number	Parent Filing Date (MM/DD/YYYY)	Parent Patent Number (if applicable)

☐ Additional U.S or PCT international application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto

As a named inventor, I hereby appoint the following registered practitioner(s) to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith:

Name	Registration Number	Name	Registration Number
Timothy W. Markison	33,534	Christopher J. Reckamp	34,414
Paul M. Anderson	39,896	Sally Daub	41,478
J. Gustav Larson	39,263		

☐ Additional registered practitioner(s) named on supplemental Registered Practitioner Information sheet PTO/SB/02C attached hereto.

Direct all correspondence to:

Markison & Reckamp, P.C.
175 West Jackson Boulevard - Suite 1015
Chicago, Illinois 60604
Telephone:312-939-9800
Facsimile: 312-939-9828

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Name of Sole or First Inventor: ☐ A petition has been filed for this unsigned inventor

Given Name (first and middle [if any])		Family Name or Surname		
Paul W.		**Campbell**		
Inventor's Signature			**Date**	7/6/99
Residence	**City: Oakland**	**State: CA**	**Country: USA**	**Citizenship: New Zealand**
Post Office Address	**6652 Dana St.**			
City: Oakland	**State: CA**	**ZIP: 94609**	**Country: USA**	

Name of Additional Joint Inventor: ☐ A petition has been filed for this unsigned inventor

Given Name (first and middle [if any])		Family Name or Surname		
Inventor's Signature			**Date**	
Residence	**City:**	**State:**	**Country:**	**Citizenship:**
Post Office Address				
City:	**State:**	**ZIP:**	**Country:**	

Name of Additional Joint Inventor: ☐ A petition has been filed for this unsigned inventor

Given Name (first and middle [if any])		Family Name or Surname		
Inventor's Signature			**Date**	
Residence	**City:**	**State:**	**Country:**	**Citizenship:**
Post Office Address				
City:	**State:**	**ZIP:**	**Country:**	

☐ Additional inventors are being named on the _____supplemental Additional Inventor(s) sheet(s) PTO/SB/02A attached hereto